## OpenSIFT+VS2017+Win11编译指南（2022.06.15）

同济大学 张林

<http://cslinzhang.gitee.io/home/>

### 基本信息

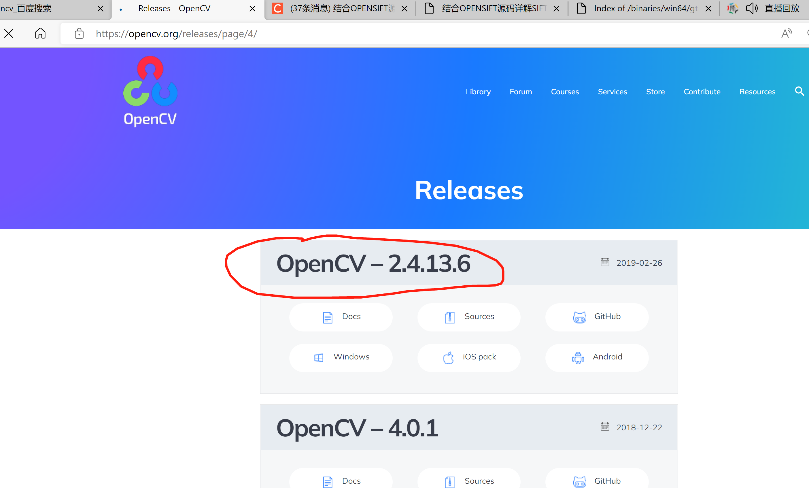
Opensift是由Rob Hess写的SIFT开源代码。OpenCV中实现的SIFT就是基于opensift实现的。本操作指南将带领大家完成在win11+VS2017环境下opensift的编译运行。

### 依赖包安装

#### 2.1 OpenCV

OpenSIFT基于比较早期的OpenCV和GTK编写的，很难移植到最新版本的OpenCV和GTK之上，主要是早期的OpenCV的数据结构在OpenCV4.0以后很多都不再支持了。我们需要安装OpenCV2.X系列的版本。

我们选择opencv2.4.13.6，可以在OpenCV官方网站下载，



下载完毕之后，解压到一个目录中即可。

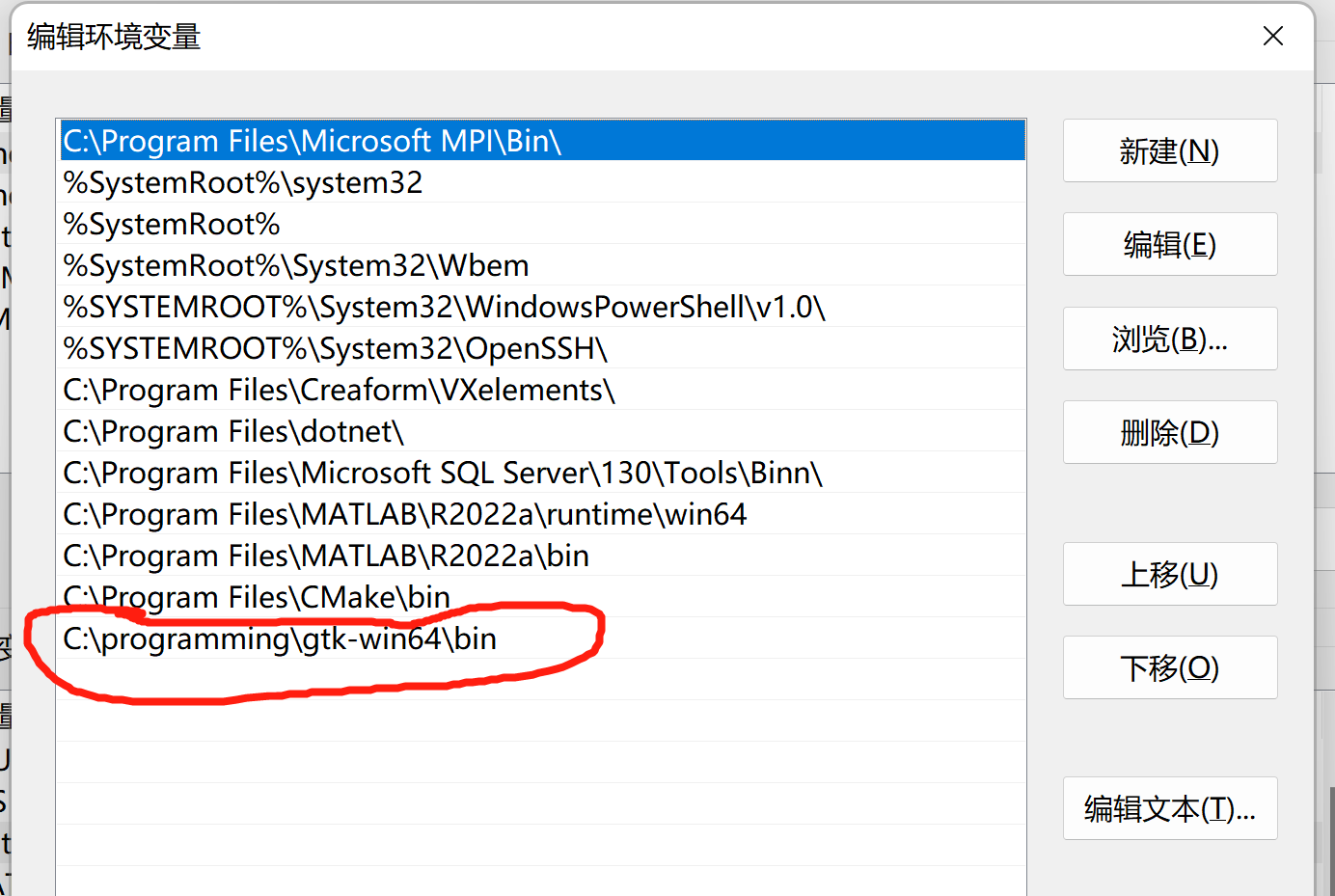
#### 2.2 GTK

GTK用2.X系列中面向x64系统的最高版本，gtk.2.22，这可以在<https://download.gnome.org/binaries/win64/> 下面下载到，



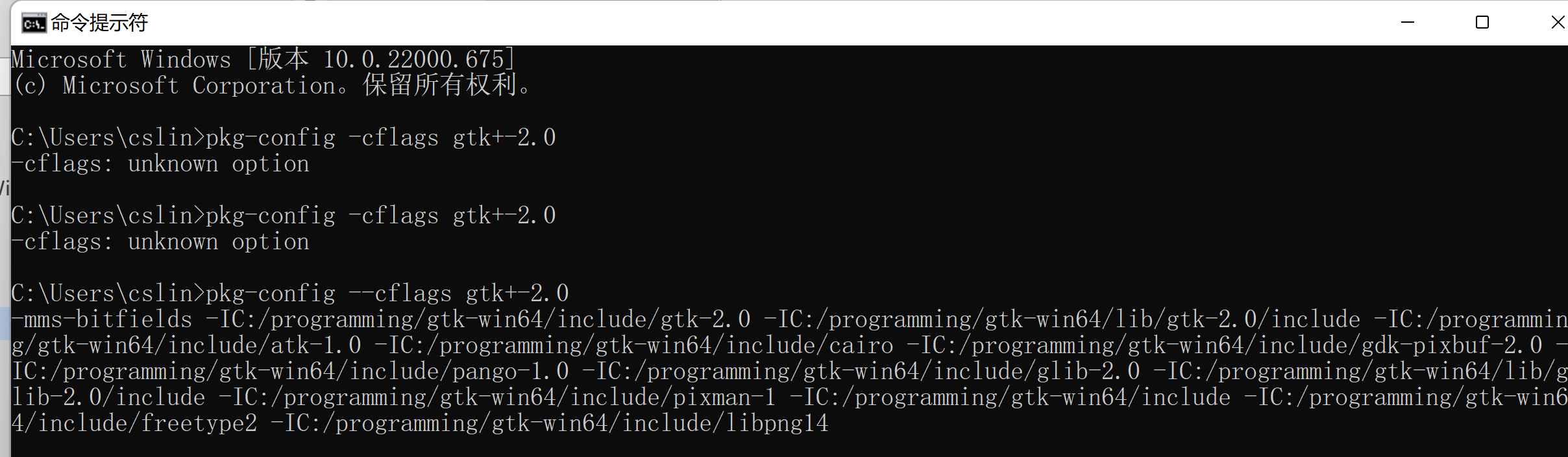
下载完毕后解压缩即可，假设解压缩之后的目录为c:\programming\gtk-win64\。

然后需要在系统环境变量path中加入gtk解压缩之后的bin目录，



Gtk安装配置完成后，我们需要测试以确认一下是否一切都正常。在cmd中输入pkg-config –cflags gtk+-2.0，会有一些输出

在cmd中输入gtk-demo命令，会出现一个gtk+的演示功能组件，证明gtk安装成功。在cmd中输入pkg-config --cflags gtk+-2.0，会有一些输出



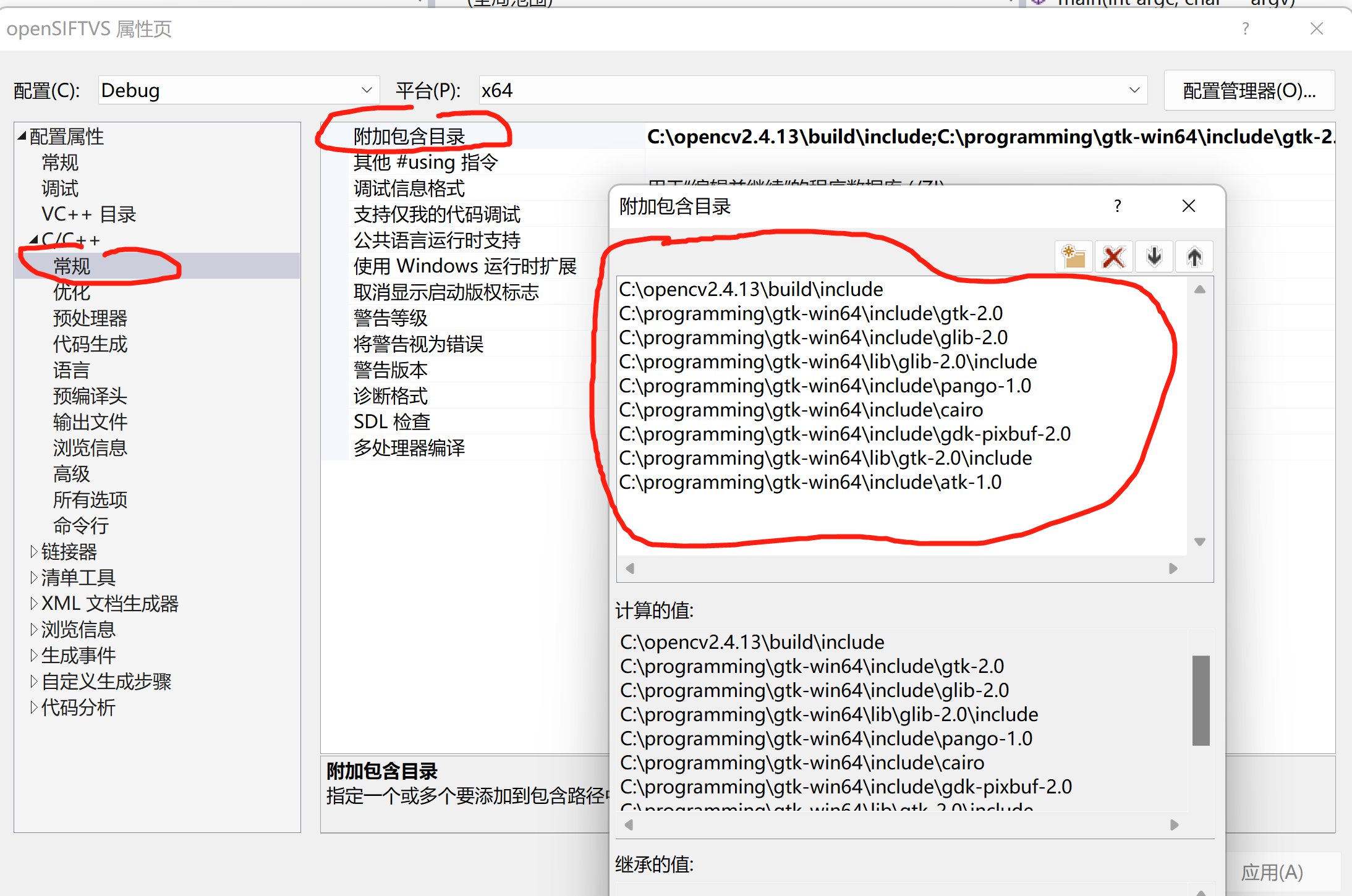
### VS项目创建

#### 3.1 创建VS项目

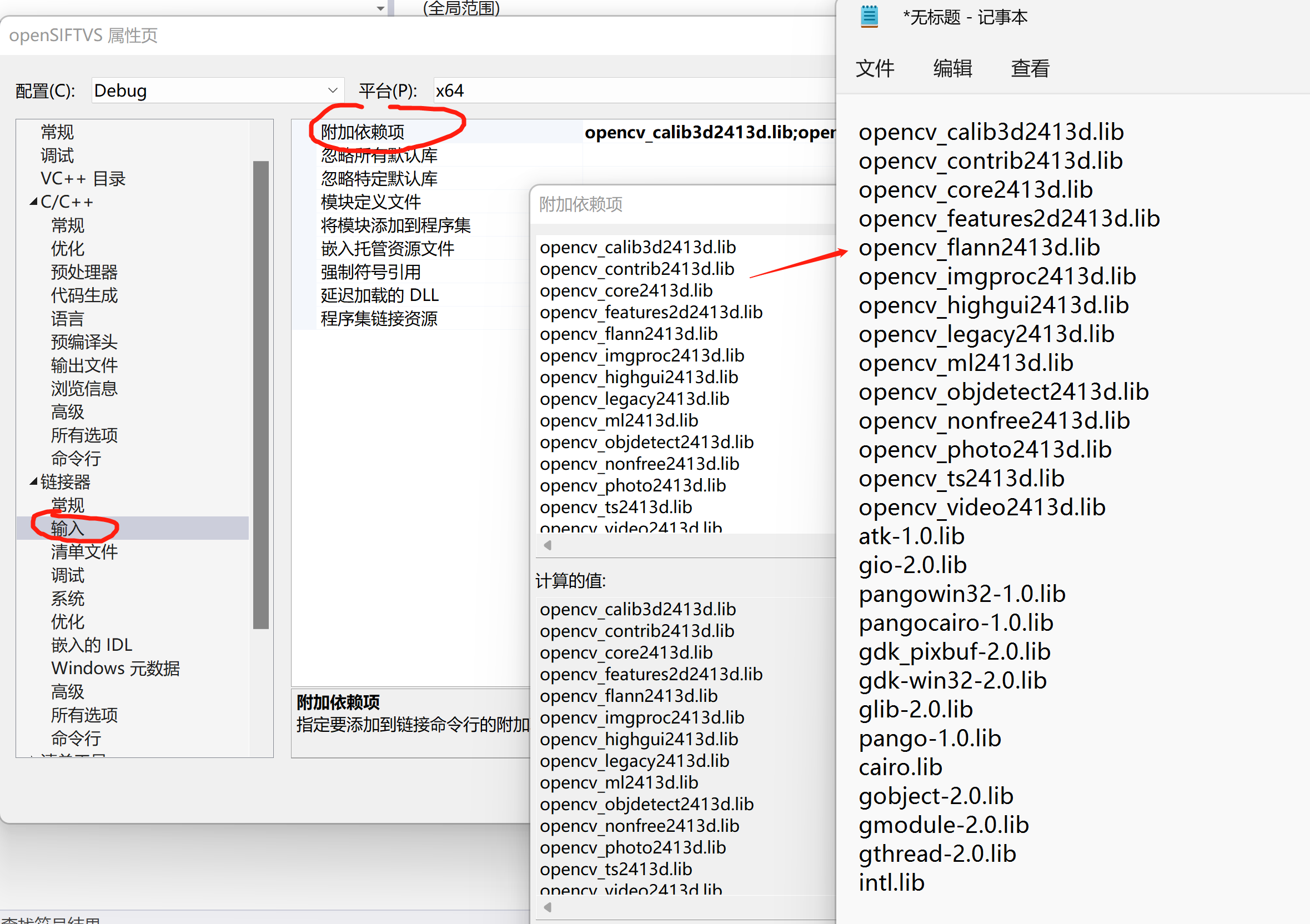
在VS中创建win32控制台程序。之后，将下载的OpenSIFT中的include和 src目录拷贝至项目目录下，在VS的解决方案[资源](http://www.haogongju.net/tag/%E8%B5%84%E6%BA%90" \t "_blank" \o "资源)管理器下，分别在头文件和源文件上右击选择添加->现有项，添加include中的h文件和src中的c文件。注意src中的match.c、dspfeat.c和siftfeat.c中都含有main函数，分别实现两张图片匹配、读取存在文件中的图像特征并显示和计算输入图像的SIFT特征，源文件中只能添加三个中的一个。我这里选择添加match.c文件。

#### 3.2 头文件与库文件添加

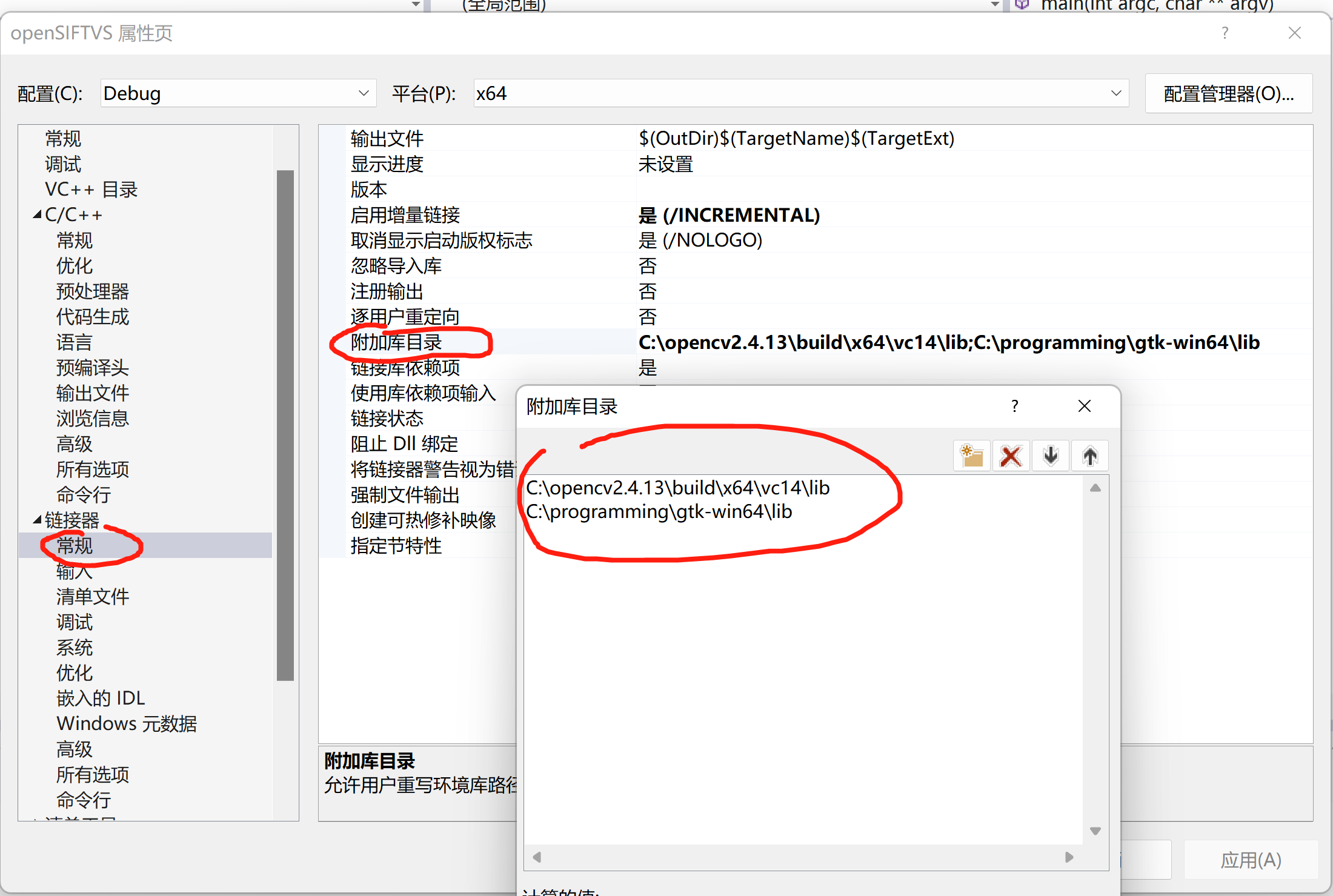
Opensift用到了opencv和GTK库，因此需要把相关的头文件目录添加至VS包含目录中，



添加相关的lib文件



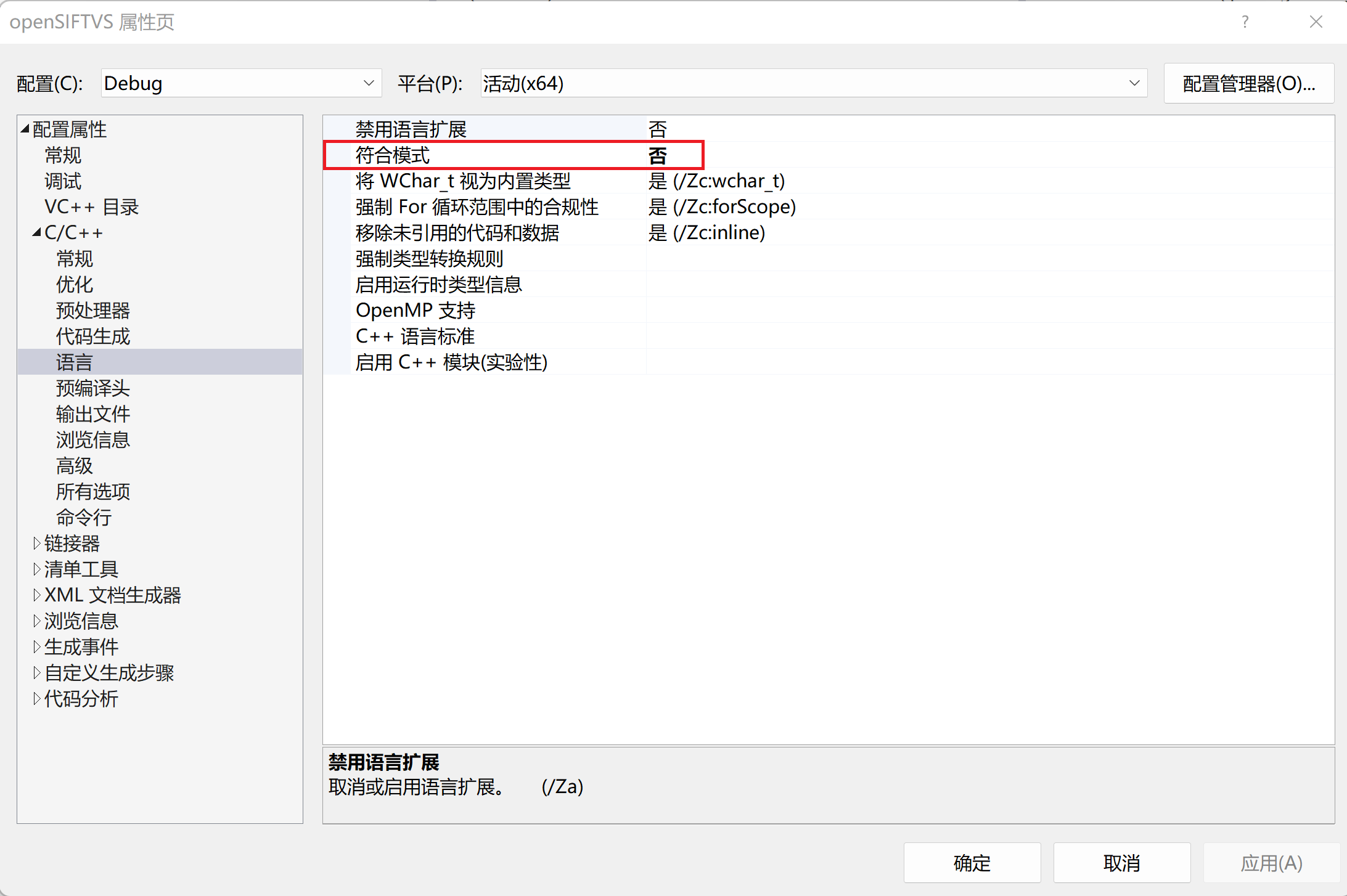
添加库文件所在路径



#### 3.3 对部分文件源码的修改

1. 编译器报出“Const char\*的实参与char\*形参不符合”错误

要按照下图所示，将“符合模式”设置为“否”。



1. “srandom”,"random”未定义

将xform.c中的srandom改为srand，random改为rand。

1. “应输入常量表达式”错误

sift.c中，应该是不支持这样实现动态数组：

const int \_intvls **=** intvls**;** double sig**[**\_intvls**+**3**],** sig\_total**,** sig\_prev**,** k**;**

这两行改为，

double **\***sig **=** **(**double **\*)**calloc**(**intvls **+** 3**,sizeof(**double**));** double sig\_total**,** sig\_prev**,** k**;**

再在最后return前加上free(sig);

1. “M\_PI未声明”错误

发生在imgfeatures.c中。在imgfeatures.h中加入：

#define M\_PI 3.14159265358979323846

1. “#include <dirent.h>”找不到

直接把这个包含注释掉即可。

1. 完成编译后，在链接阶段会报出“\_sift\_features”等无法解析的外部符号的错误

这是由于c++和c混编所引起的。在出现extern的头文件中，在最早出现extern的语句前面加上，

#ifdef \_\_cplusplus

extern "C" {

#endif

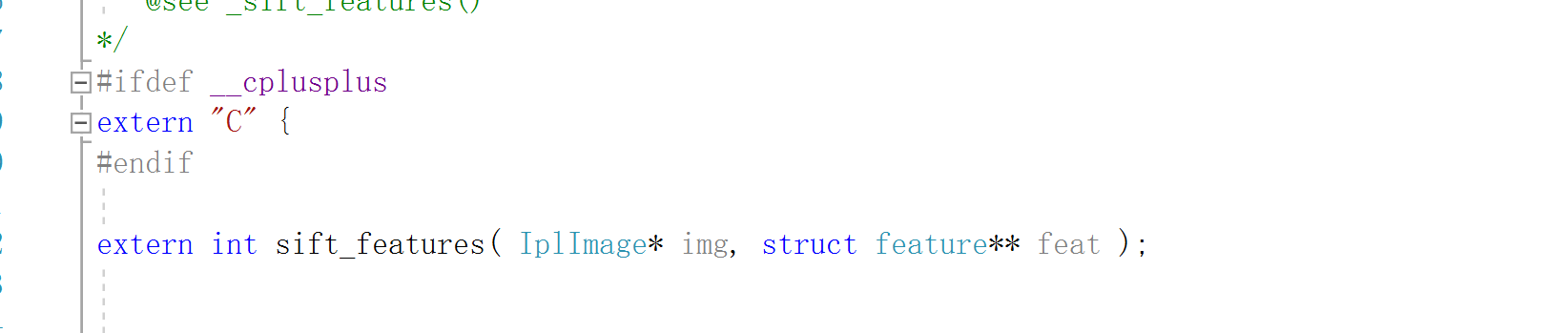
在文件最后的#endif前面加上，

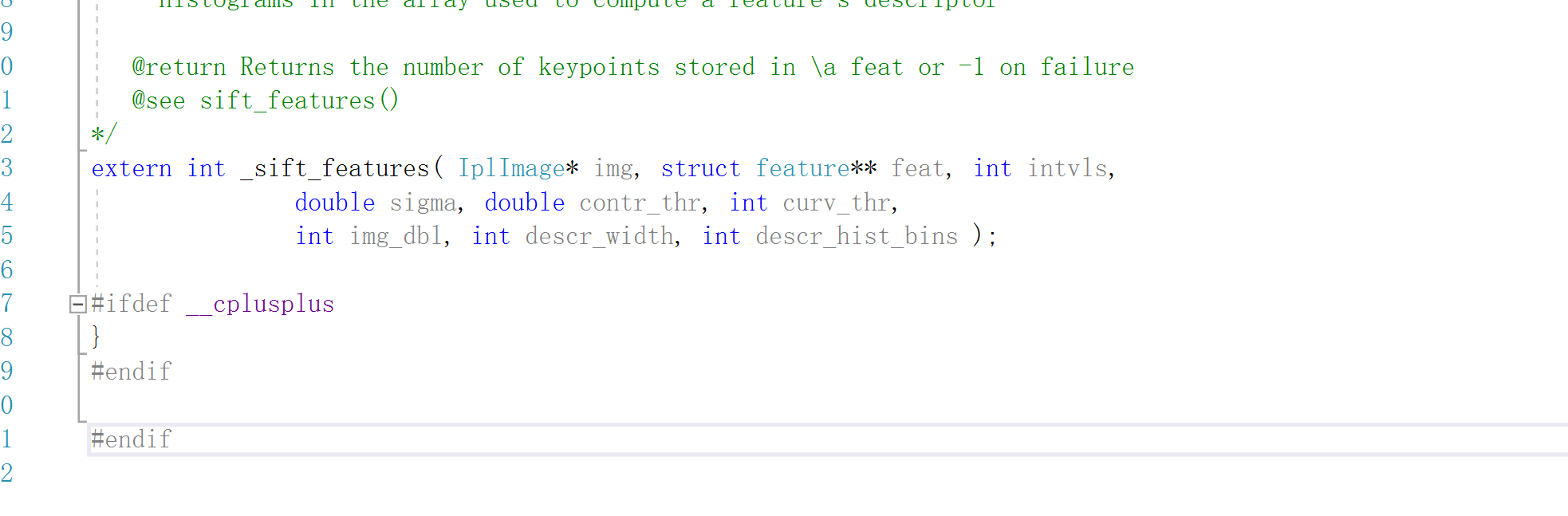
#ifdef \_\_cplusplus

}

#endif

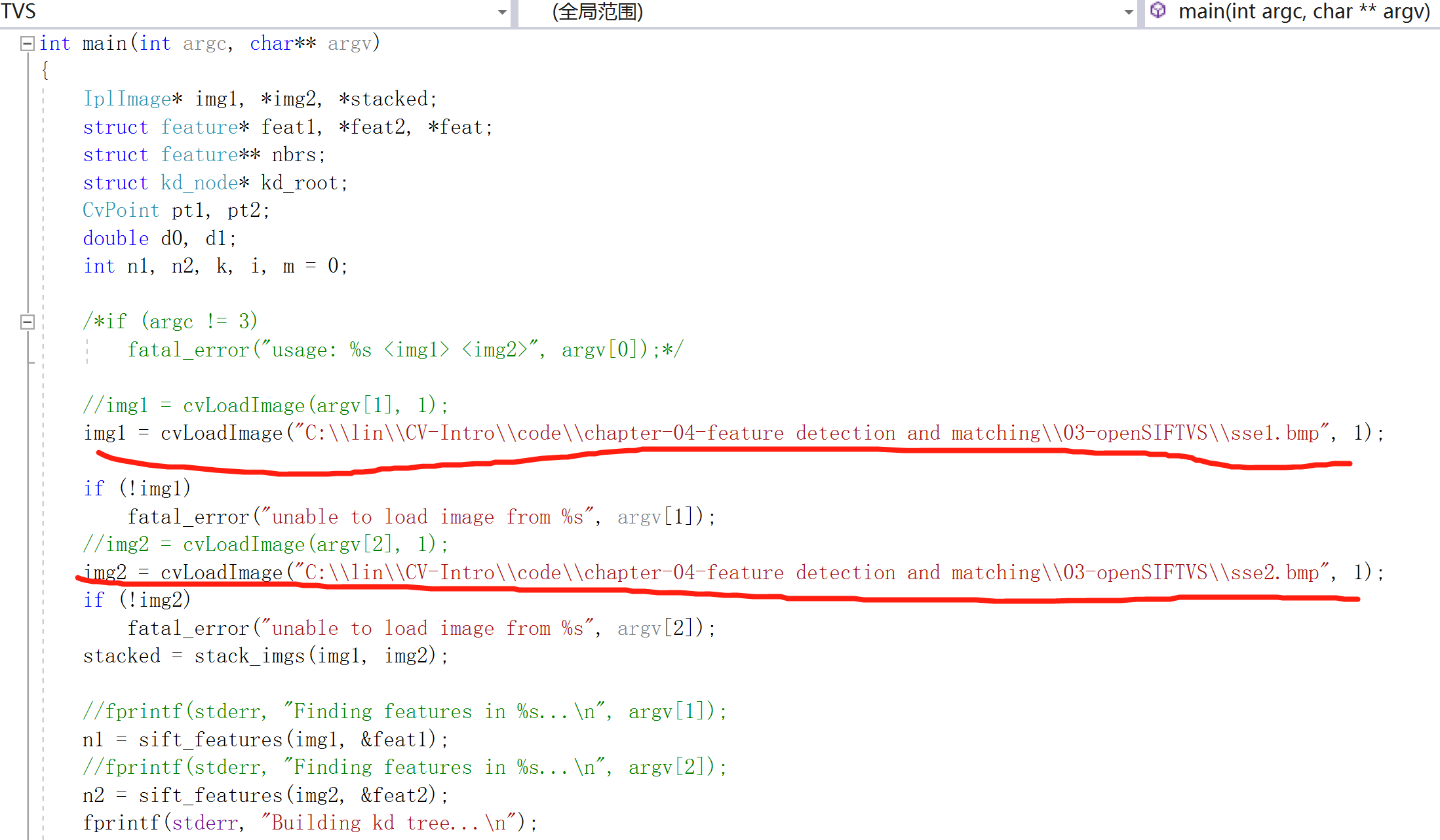
最终形如，



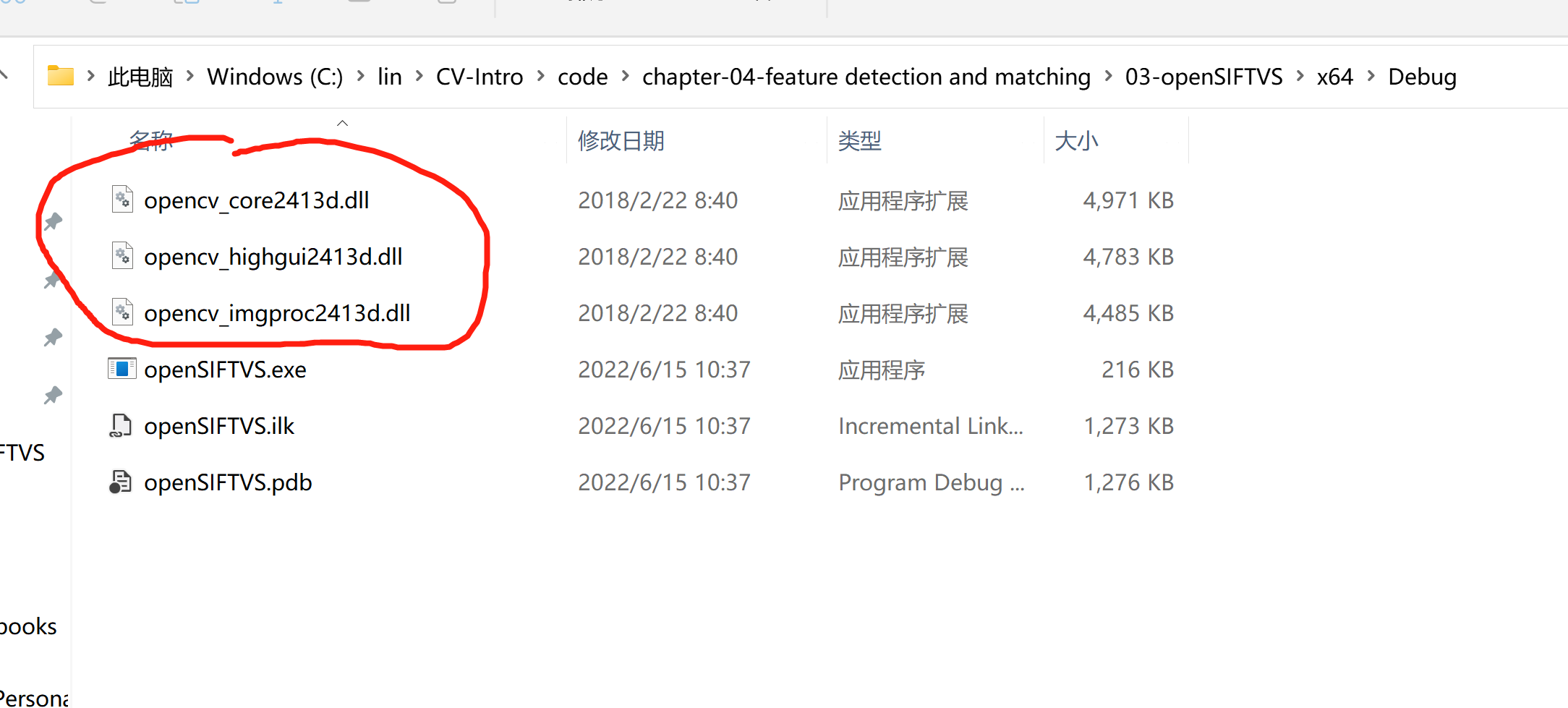


1. 修改main函数

由于是测试用途，为了方便，可以把两幅待匹配的图像路径直接硬编码在代码中，



1. 相关dll文件拷贝到编译输出目录下，并运行程序



程序运行后，会显示如下的SIFT匹配结果，

